

USING MYSQL JOINS

<http://www.tutorialspoint.com/mysql/mysql-using-joins.htm>

Copyright © tutorialspoint.com

Thus far we have only been getting data from one table at a time. This is fine for simple takes, but in most real world MySQL usage you will often need to get data from multiple tables in a single query.

You can use multiple tables in your single SQL query. The act of joining in MySQL refers to smashing two or more tables into a single table.

You can use JOINS in SELECT, UPDATE and DELETE statements to join MySQL tables. We will see an example of LEFT JOIN also which is different from simple MySQL JOIN.

Using Joins at Command Prompt:

Suppose we have two tables **tcount_tbl** and **tutorials_tbl** in TUTORIALS. A complete listing is given below:

Example:

Try out following examples:

```
root@host# mysql -u root -p password;
Enter password:*****
mysql> use TUTORIALS;
Database changed
mysql> SELECT * FROM tcount_tbl;
+-----+-----+
| tutorial_author | tutorial_count |
+-----+-----+
| mahran         | 20             |
| mahnaz         | NULL           |
| Jen            | NULL           |
| Gill           | 20             |
| John Poul      | 1              |
| Sanjay         | 1              |
+-----+-----+
6 rows in set (0.01 sec)
mysql> SELECT * from tutorials_tbl;
+-----+-----+-----+-----+
| tutorial_id | tutorial_title | tutorial_author | submission_date |
+-----+-----+-----+-----+
| 1           | Learn PHP     | John Poul      | 2007-05-24      |
| 2           | Learn MySQL   | Abdul S        | 2007-05-24      |
| 3           | JAVA Tutorial | Sanjay         | 2007-05-06      |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)
mysql>
```

Now we can write a SQL query to join these two tables. This query will select all the authors from table **tutorials_tbl** and will pickup corresponding number of tutorials from **tcount_tbl**.

```
mysql> SELECT a.tutorial_id, a.tutorial_author, b.tutorial_count
-> FROM tutorials_tbl a, tcount_tbl b
-> WHERE a.tutorial_author = b.tutorial_author;
+-----+-----+-----+
| tutorial_id | tutorial_author | tutorial_count |
+-----+-----+-----+
| 1           | John Poul      | 1              |
| 3           | Sanjay         | 1              |
+-----+-----+-----+
2 rows in set (0.01 sec)
mysql>
```

Using Joins in PHP Script:

You can use any of the above mentioned SQL query in PHP script. You only need to pass SQL query into PHP function `mysql_query()` and then you will fetch results in usual way.

Example:

Try out following example:

```
<?php
$dbhost = 'localhost:3036';
$dbuser = 'root';
$dbpass = 'rootpassword';
$conn = mysql_connect($dbhost, $dbuser, $dbpass);
if(! $conn )
{
    die('Could not connect: ' . mysql_error());
}
$sql = 'SELECT a.tutorial_id, a.tutorial_author, b.tutorial_count
      FROM tutorials_tbl a, tcount_tbl b
      WHERE a.tutorial_author = b.tutorial_author';

mysql_select_db('TUTORIALS');
$retval = mysql_query( $sql, $conn );
if(! $retval )
{
    die('Could not get data: ' . mysql_error());
}
while($row = mysql_fetch_array($retval, MYSQL_ASSOC))
{
    echo "Author:{$row['tutorial_author']} <br> ".
        "Count: {$row['tutorial_count']} <br> ".
        "Tutorial ID: {$row['tutorial_id']} <br> ".
        "-----<br>";
}
echo "Fetched data successfully\n";
mysql_close($conn);
?>
```

MySQL LEFT JOIN:

A MySQL left join is different from a simple join. A MySQL LEFT JOIN gives extra consideration to the table that is on the left.

If I do a LEFT JOIN, I get all records that match in the same way and IN ADDITION I get an extra record for each unmatched record in the left table of the join - thus ensuring (in my example) that every AUTHOR gets a mention:

Example:

Try out following example to understand LEFT JOIN:

```
root@host# mysql -u root -p password;
Enter password:*****
mysql> use TUTORIALS;
Database changed
mysql> SELECT a.tutorial_id, a.tutorial_author, b.tutorial_count
      -> FROM tutorials_tbl a LEFT JOIN tcount_tbl b
      -> ON a.tutorial_author = b.tutorial_author;
+-----+-----+-----+
| tutorial_id | tutorial_author | tutorial_count |
+-----+-----+-----+
| 1 | John Poul | 1 |
| 2 | Abdul S | NULL |
| 3 | Sanjay | 1 |
+-----+-----+-----+
```

```
3 rows in set (0.02 sec)
```

You would need to do more practice to become familiar with JOINS. This is a but complex concept in MySQL/SQL and will become more clear while doing real examples.